

Studio teorico dell'adsorbimento di molecole semplici su superfici metalliche

G.F. Tantardini

Università degli Studi di Milano, Dip. di Chimica Fisica ed Elettrochimica

Abstract

Viene descritta la realizzazione di una nuova versione di un programma di dinamica molecolare reattiva. Il precedente codice sequenziale è stato parallelizzato, utilizzando l'ambiente operativo PVM, e portato su un cluster di workstation HP-META, ottenendo un miglioramento delle prestazioni di calcolo di un ordine di grandezza.

L'attività è stata svolta nell'ambito del Centro di Modellistica Computazionale.

Sul sistema Convex META-HP sono stati effettuati calcoli intensivi di dinamica molecolare reattiva utilizzando il programma TRAJ, originariamente scritto per workstation [1]. Il programma consente di effettuare simulazioni di scattering reattivo di fasci di molecole in fase gassosa su superfici metalliche. L'obiettivo delle simulazioni è quello di ottenere informazioni teoriche da confrontare con dati provenienti da esperimenti con fasci molecolari supersonici adsorbiti su superfici metalliche dotate di elevata purezza e regolarità. Tali esperimenti consentono di approfondire in modo determinante le conoscenze dei fenomeni che avvengono nelle reazioni catalizzate da superfici metalliche. Le simulazioni si basano sull'uso di equazioni classiche (hamiltoniane, newtoniane) per descrivere il moto delle molecole del fascio e di potenziali quantistici per descrivere l'interazione tra la molecola e la superficie.

Quando la superficie viene considerata rigida e non è permesso alcun scambio di energia tra la molecola incidente e gli atomi della superficie metallica, è possibile effettuare simulazioni statisticamente significative anche su workstation medio-piccole. Poiché la probabilità di adsorbimento della molecola gassosa (coefficiente di sticking) viene calcolata impiegando il metodo Monte Carlo che, come è noto, converge molto lentamente, è necessario

in alcuni casi allargare la base statistica delle simulazioni, soprattutto quando si è in presenza di eventi poco frequenti, con conseguente notevole aumento dei tempi di calcolo e di attesa. Pertanto, in questi casi è stato necessario poter disporre di sistemi massicciamente paralleli come il sistema META-HP in cui il CONVEX C3820 gestisce, come front-end, otto slaves costituiti da computers HP/735 attraverso il sistema NQS (*Network Queuing System*). Quindi, in questa prima fase, il lavoro è stato limitato alla semplice conversione del programma TRAJ alle caratteristiche dei nuovi computers. Sul sistema META-HP, sono stati effettuati calcoli dinamici per il chemisorbimento di idrogeno molecolare sulle superfici Cu(111) [2] e su Ni(111) [3]. Sono stati anche effettuati calcoli di dinamica quantistica con il programma CCWP basato sul metodo del pacchetto d'onda [4], con ottimi risultati per quanto riguarda i tempi di risposta e le risorse di memoria RAM.

Si è visto che risulta sempre più importante poter studiare le conseguenze che lo scambio di energia tra la molecola e gli atomi della superficie (*energy accommodation*) può avere sull'adsorbimento. Ciò è stato fatto utilizzando equazioni di Langevin generalizzate nella formulazione Ghost Atoms [3] o semplici oscillatori di Langevin [5] per descrivere il moto degli atomi della superficie. L'aumento dei gradi

di libertà del problema comporta un rapido aumento del numero di equazioni differenziali da integrare e quindi i tempi di calcolo aumentano di alcuni ordini di grandezza. Nella seconda fase della ricerca, si è quindi passati alla ristrutturazione del codice TRAJ in modo tale da poter gestire le risorse del cluster di 8 HP/735 in modo parallelo con il software PVM (*Parallel Virtual Machine*). A tal scopo, è stato costruito un codice in grado di funzionare come master, come slave e come programma per computer non parallelo. Il master è stato posto sull'HP/735 n.3 e gli slaves su tutti gli HP, compreso HP-3. Il master prepara le condizioni iniziali di tutte le traiettorie che distribuisce agli slaves (una traiettoria per slave) e rimane in attesa che uno slave comunichi di aver terminato i calcoli della traiettoria assegnatagli. Il master riceve i risultati del calcolo effettuato da quello slave, li memorizza su un file e assegna allo slave una nuova traiettoria da calcolare e così fino ad esaurimento. Lo slave, da parte sua, riceve le condizioni iniziali della traiettoria da calcolare, effettua il calcolo, comunica i risultati al master e si prepara a ricevere i dati iniziali di una nuova traiettoria. Quando tutte le traiettorie sono state calcolate dagli slaves, il master procede alla lettura dei dati memorizzati su file, esegue la statistica sul set di traiettorie calcolate e scrive i risultati riassuntivi della simulazione. Il programma TRAJ non effettua chiamate dirette alle routines della libreria PVM Fortran, ma utilizza una subroutine di interfaccia chiamata pvm (vedi allegato) con diverse entries che si occupa di definire e riempire i buffers di comunicazione, di inviarli dal master allo slave o viceversa, di riceverli e di prelevare le informazioni. Per l'utilizzo del programma TRAJ su un computer dotato di una sola CPU è sufficiente banalizzare la subroutine pvm eliminando tutte le chiamate alle routines della libreria PVM.

In questo modo, sul cluster HP/735 è stato possibile ridurre di un ordine di grandezza i tempi di calcolo e di ottenere, in tempi molto più brevi, i risultati di una simulazione. Questa forte riduzione del tempo di risposta consente da un lato di aumentare convenientemente la base statistica della simulazione aumentando il numero di molecole (traiettorie) che costituiscono il fascio simulato e dall'altro di aumentare il numero degli atomi della superficie che si muovono per una trattazione più realistica dello

scambio di energia tra molecola incidente e superficie. Inoltre, rende possibile migliorare il potenziale di interazione tra la molecola e la superficie. Per tutto questo, anche il sistema cluster HP/735 si rivela insufficiente e sarà necessario puntare alla realizzazione di una versione del programma TRAJ in grado di sfruttare le risorse di EXEMPLAR che è costruito da 32 processori, sempre utilizzando l'approccio PVM alla parallelizzazione che consente ottimi risultati senza eccessivo dispendio di energie nella ristrutturazione del codice di calcolo. La versione PVM di TRAJ è stata e viene utilizzata per lo studio del fisi- e chemi-adsorbimento di ossigeno sulla superficie Ag(110) che riveste particolare interesse per la comprensione della reazione di epossidazione dell'ossigeno molecolare su catalizzatori di argento.

Nella riscrittura del codice TRAJ per l'utilizzo del software PVM è stato essenziale l'aiuto del Dr. Giampaolo Bottoni che ci ha assistito fornendoci esempi semplici e completi di colloquio master-slave in ambiente PVM.

Pubblicazioni

- [1] G.F. Tantardini, M. Simonetta, "*Quasi-classical trajectory study of the chemisorption of H₂ on the Pt(111) surface*" - Surf. Sci. 105, 517-535 (1981);
- G.F. Tantardini, M. Simonetta, "*Quasi-classical trajectories study of the chemisorption of H₂ on the Pt(111) surface: effect of surface motion*" - Chem. Phys. Letters 87, 420-425 (1982)
- G.F. Tantardini, "*The Chemisorption Dynamics of Hydrogen on Metal Surfaces*" in: "*Cluster Models for Surface and Bulk Phenomena*", NATO ASI Series B, Eds. G. Pacchioni, P. Bagus and F. Parmigiani (Plenum Press, New York), B 283, 389-404 (1992)
- [2] L. Pesce, G.F. Tantardini, "*Wave Packet Simulations of the Physisorption of Molecular*

Hydrogen on the (001) Surface of Copper" Il Vuoto. Scienza e Tecnologia, 1995, in stampa

[3] A. Forni, G. Wiesenekker, E.J. Baerends, G.F. Tantardini, "A Dynamical Study of the Chemisorption of Molecular Hydrogen on the Cu(111) Surface", J. Phys. Cond Matter 7 (1995) 1

[4] A. Forni and G.F. Tantardini, "A Simulation Study of the Chemisorption Dynamics of Molecular Hydrogen on the Ni(111) Surface", Surf. Sci., 1996, in stampa

[5] I. Pazzi, G.F. Tantardini, "Dynamical Simulations Study of the Adsorption of Oxygen on the (110) Surface of Silver", Il Vuoto. Scienza e Tecnologia, 1996, in corso di pubblicazione

N.B.: Negli acknowledgements dei lavori [2], [3], [4], [5] è stato fatto esplicito riferimento ai progetti del Centro di Modellistica Computazionale.

Comunicazioni a Congressi

[C1] L. Pesce, G.F. Tantardini, "Wave Packet Simulations of the Physisorption of Molecular Hydrogen on the (001) Surface of Copper", 19th Annual Meeting on Advances in Surface and Interface Physics, Modena, 19-21 dicembre 1994

[C2] I. Pazzi, G.F. Tantardini, "Adsorption of Oxygen on Ag(110): A Dynamical Study", 15-th European Conference on Surface Science, Lille (Francia), 4-8 settembre 1995

[C3] A. Forni, G.F. Tantardini, "Chemisorption Dynamics of Hydrogen on the Ni(111) Surface", 15-th European Conference on Surface Science, Lille (Francia), 4-8 settembre 1995

[C4] I. Pazzi, G.F. Tantardini, "Dynamical Simulations Study of the Adsorption of Oxygen on Ag(110)", 20th Annual Meeting on Advances in Surface and Interface Physics, Modena, 18-20 dicembre 1995

Subroutine di interfaccia tra il programma TRAJ e le librerie PVM

```
subroutine pvm
c   il programma di dinamica reattiva
c   TRAJ non effettua chiamate dirette alla
c   libreria PVM
c   Fortran, ma utilizza questa subroutine
c   come interfaccia
common ./cpvm/ ipvm, ntask,
numt
common /a/ il, ...
common /b/ itl, ...
common /c/ ifl, ...
dimension itids(100)
save
return
```

```
entry pvmcheck
c   con questa chiamata si stabilisce se si
c   tratta di processo master o slave
c   ipvm = 1 il processo è master
c   ipvm = 2 il processo è slave
call pvmfparent (iptid)
if (iptid.lt.0) then
    ipvm = 1
else
    ipvm = 2
endif
return
```

```
entry pvmstart
c   con questa chiamata il master attiva
c   sugli slaves ntask processi di nome traj
c   numt è il numero di processi
c   effettivamente attivati
call pvmfspawn
('traj',0, '.',ntask,itids,numt)
return
```

```
entry pvmfine
c   con questa chiamata il programma
c   TRAJ esce da PVM
do j=1,numt
    call pvmfkill (itids(j),info)
enddo
call pvmfexit (info)
return
```

```
entry pvminit
c   il master spedisce a ciascun slave i dati
c   necessari per il calcolo di tutte le
c   traiettorie
```

```

call pvmfinitssend (0,info)
call pvmfpack (3,il,8,1,info)
call pvmfpack (...)
-----
call pvmfmcast
(numt,itids,100,info)
return

```

```

call pvmfunpack
(3,ifl,60,1,info)
call pvmfunpack (...)
-----
return
end

```

```

      entry pvminvia
c    il master trasmette allo slave le
      condizioni iniziali della singola
      traiettoria
      call pvmfinitssend (0,info)
      call pvmfpack
      (3,itl,624,1,info)
      call pvmfpack (...)
      -----
      call pvmfssend (itid,200,info)
      return

```

```

      entry pvminits
c    lo slave riceve dal master i dati iniziali
      per il calcolo di tutte le traiettorie
      call pvmfrecv (iptid,100,info)
      call pvmfunpack (3,il,8,1,info)
      call pvmfunpack (...)
      -----
      return

```

```

      entry pvmriceves
c    lo slave riceve dal master le condizioni
      iniziali della singola traiettoria
      call pvmfrecv
      (iptid,200,ibufid)
      call pvmfunpack
      (3,itl,624,1,info)
      call pvmfunpack (...)
      -----
      return

```

```

      entry pvminvias
c    lo slave trasmette al master i dati finali
      della singola traiettoria
      call pvmfinitssend (0,info)
      call pvmfpack (3,ifl,60,1,info)
      call pvmfpack (...)
      -----
      call pvmfssend (iptid,300,info)
      return

```

```

      entry pvmriceve
c    il master riceve dallo slave i dati finali
      della singola traiettoria
      call pvmfrecv (-1,300,ibufid)

```